情報検索とのマルチタスク学習による大規模機械読解

西田 京介　　斉藤 いつみ　　大塚 淳史　　浅野 久子　　富田 準二
日本電信電話株式会社 NTT メディアインテリジェンス研究所
nishida.kyosuke@lab.ntt.co.jp

1 はじめに

人工知能がテキストを読み解いて質問に答える機械読解が近年注目を集めている．SQuAD [13] を始めとした大規模かつ高品質なデータセットに基いた深層学習により，最近では人間に匹敵する質問応答精度が得られている．その一方で，最新のモデル [14, 17] は，質問に答える際に参照する 1 つのテキスト中に回答が含まれていることを仮定するため，複数のテキストを知識源とした質問応答を行うことは出来ない．

そこで本研究では，複数の非構造化テキストに対して機械読解を行う**大規模機械読解**に取り組む．大規模機械読解は Chen et al. により提唱された新しいタスクであり，質問文に含まれるキーワードによる情報検索と，ニューラルネットワークによる機械読解を単純に連結することで実現された [2]．しかし，単純なキーワード検索では，パッセージ中に質問に回答するために必要な情報が含まれているかを識別することができないため，情報検索の精度がボトルネックであった．

ここで，パッセージ中から回答文字列の範囲を特定する機械読解の能力は，質問に回答可能なパッセージか否かを識別する情報検索の精度向上に貢献すると考えられる．しかし，機械読解モデルは質問に適合するパッセージのみで学習を行うため，情報検索へ単純に転移しても有効ではない．高精度に大規模機械読解を行うためには，情報検索と機械読解の両方の能力を持つようにモデルを学習することが重要と考える．そこで本研究では，情報検索と機械読解のマルチタスク学習を行うモデルを提案する．評価実験により，提案手法は大規模機械読解において state-of-the-art の性能を持つことを示す．

2 問題定義

問題 1 (**大規模機械読解**). パッセージ集合 D の中から，入力された質問 q に適合する k 個のパッセージを検索し（情報検索タスク），次に，検索結果のパッセージから回答を抽出する（機械読解タスク）．

定義 1. **質問**は，自然言語で記述された文とする．

定義 2. **パッセージ**は，短い部分文書（数百単語程度）であり，画像などの非テキスト情報は含まない．

定義 3. **回答**は，パッセージ中の任意の長さの文字範囲とする．回答は単語や固有名詞に限定されず，任意のフレーズとする．また，回答はパッセージ中から抽出されるものとし，生成はされない．

定義 4. **適合パッセージ**とは，質問に回答するために必要な情報がすべて含まれたパッセージとする．

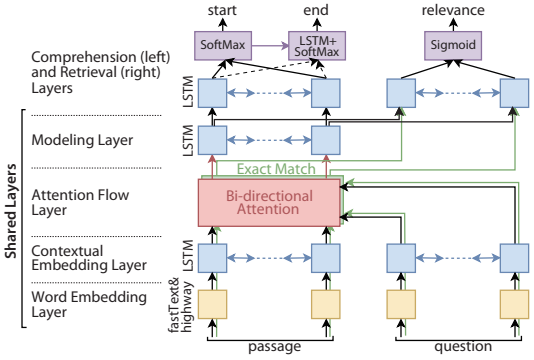

図 1: 読解と検索を行う大規模機械読解モデル．

3 提案モデル

提案モデルは，機械読解において最先端のモデルの 1 つである BiDAF [14] をベースとし，情報検索の層を追加したものである（図 1）．6 層のうち最初の 4 層を両タスクで共有することで，情報検索を高精度に行う．

3.1 単語埋め込み層

$x = [x_1, \ldots, x_T]^\top$ と $q = [q_1, \ldots, q_J]^\top$ を入力されたパッセージと質問に含まれる各単語の one-hot 表現の系列とする．単語埋め込み層は，各 one-hot 表現（語彙数 V）を v 次元の連続値空間へ重み行列 $W^e \in \mathbb{R}^{v \times V}$ により射影する．そして，2 層の highway network [15] を用いて，パッセージ（単語数 T）と質問（単語数 J）について v 次元ベクトルの系列 $X \in \mathbb{R}^{v \times T}$ と $Q \in \mathbb{R}^{v \times J}$ を出力する．提案モデルでは，単語ベクトルとしてサブワードを考慮することで OOV に対しても適切な埋め込み表現が得られる fastText [1] を利用する．

3.2 コンテキスト埋め込み層

単層の双方向 LSTM（各方向に d 次元の状態を持つ）を用いて，パッセージの単語ベクトル系列 X から $H \in \mathbb{R}^{2d \times T}$ を，質問の単語ベクトル系列 Q から $U \in \mathbb{R}^{2d \times J}$ をそれぞれ出力する．

3.3 アテンション層

パッセージから質問，および，質問からパッセージの双方向のアテンションを計算する．パッセージの t 番目の単語と，質問の j 番目の単語の類似度

$$S_{tj} = w^{s\mathsf{T}}[H_t; U_j; H_t \odot U_j] \tag{1}$$

を要素に持つ類似度行列 $S \in \mathbb{R}^{T \times J}$ を求める．ここで，$w^s \in \mathbb{R}^{6d}$ は学習パラメータ，$\odot$ はアダマール積，$[;]$ は行方向に沿ったベクトル連結の演算子を表す．

パッセージから質問へのアテンションは，パッセージの単語ごとに質問文の重要な単語へ重みを付けてプーリングベクトル $\tilde{U}_t = \sum_j a_{tj} U_j \in \mathbb{R}^{2d}$ を計算する．ここで，$a_t = \mathrm{softmax}_j(S_t) \in \mathbb{R}^J$ である．

質問からパッセージへのアテンションは，質問文のいずれかの単語に強く関連する単語に重みを付けてプーリングベクトル $\tilde{h} = \sum_t b_t H_t \in \mathbb{R}^{2d}$ を求める．ここで，$b = \mathrm{softmax}_t(\max_j(S)) \in \mathbb{R}^T$ である．そして，$\tilde{h}$ を T 回列方向に並べ，$\tilde{H} \in \mathbb{R}^{2d \times T}$ を得る．

双方向アテンションは，質問に基づいたパッセージ単語の表現としてベクトル系列 G を計算する．

$$G = [H; \tilde{U}; H \odot \tilde{U}; H \odot \tilde{H}] \in \mathbb{R}^{8d \times T} \tag{2}$$

3.4 モデリング層

G を入力とし，単層の双方向 LSTM を作用させ $M \in \mathbb{R}^{2d \times T}$ を出力する．M は読解・検索層への入力となる．

3.5 読解層

機械読解タスクは質問に対する応答となるフレーズの抽出が目的であり，このフレーズはパッセージ中の開始および終了単語位置から予測される．

まず，読解層では単層の双方向 LSTM に M を入力として与えて $M^1 \in \mathbb{R}^{2d \times T}$ を獲得し，この M^1 に基づいて開始位置の確率分布 p^1 を計算する．

$$p^1 = \mathrm{softmax}_t(w^{1\mathsf{T}}[G; M^1]) \ \in \mathbb{R}^T \tag{3}$$

ここで，$w^1 \in \mathbb{R}^{10d}$ は学習パラメータである．

次に，pointer networks [16] のアイデアに基いて，開始位置を条件とした終了位置の予測を行う．まず，$\tilde{m}^1 = \sum_t p_t^1 M_t^1 \in \mathbb{R}^{2d}$ を求め，これを T 回列方向に並べ $\tilde{M}^1 \in \mathbb{R}^{2d \times T}$ を得る．そして，$[G; M^1; \tilde{M}^1; M^1 \odot \tilde{M}^1] \in \mathbb{R}^{14d \times T}$ を別の単層双方向 LSTM に渡し，$M^2 \in \mathbb{R}^{2d \times T}$ を求める．

最後に，終了位置の確率分布 p^2 を計算する．

$$p^2 = \mathrm{softmax}_t(w^{2\mathsf{T}}[G; M^2]) \in \mathbb{R}^T \tag{4}$$

ここで，$w^2 \in \mathbb{R}^{10d}$ は学習パラメータである．

3.6 検索層

検索層は情報要求である質問に適合するパッセージの識別を目的とし，モデリング層の出力 M を適合度スコアに変換する．

まず，質問とパッセージ単語の完全一致について考慮するために，パッセージの各単語が，質問単語のいずれかと完全一致する場合に 1，否の場合に 0 となるベクトル $\tilde{B} \in \mathbb{R}^{1 \times T}$ を求める．次に，$[M; \tilde{B}] \in \mathbb{R}^{(2d+1) \times T}$ を，単層の双方向 LSTM に渡して $M^r \in \mathbb{R}^{2d \times T}$ を得る．この M^r について検索に役立つ単語表現へ重みを付けたアテンションプーリングを取る．

$$\tilde{m}^r = \sum\nolimits_t \beta_t M_t^r \ \in \mathbb{R}^{2d} \tag{5}$$

この重み $\beta \in \mathbb{R}^T$ は，質問に基づくパッセージ単語の表現とコンテキストベクトル w^c の内積から得られる．

$$\beta_t = \exp({m_t}^{\mathsf{T}} w^c) / \sum\nolimits_{t'} \exp({m_{t'}}^{\mathsf{T}} w^c) \tag{6}$$

ここで，$m_t = W^a M_t^r + b^a$ である．$W^a \in \mathbb{R}^{c \times 2d}$ と $b^a, w^c \in \mathbb{R}^c$ は学習パラメータである．最後に，質問に対するパッセージの適合度 p^r を出力する．

$$p^r = \mathrm{sigmoid}({w^r}^{\mathsf{T}} \tilde{m}^r) \ \in [0, 1] \tag{7}$$

ここで，$w^r \in \mathbb{R}^{2d}$ は学習パラメータである．

3.7 マルチタスク学習

情報検索と機械読解タスクの損失を結合した $L(\theta) = L_{RC} + \lambda L_{IR}$ を最小化する．ここで，θ はニューラルネットワークすべての学習パラメータであり，λ は調整用のハイパーパラメータである．提案モデルは，訓練データセット中の質問，パッセージ，回答範囲の組を正例とし，負例は訓練データセットから生成する．なお，負例の生成方法は 4.4 節にて説明する．

情報検索タスクの損失 L_{IR} は，2 値のクロスエントロピーにより求める．

$$L_{IR} = -\frac{1}{N} \sum\nolimits_i \left(r_i \log p^r + (1 - r_i) \log(1 - p^r)\right) \tag{8}$$

ここで，N は訓練サンプル数，r_i は i 番目のサンプルが正例のとき 1，負例のとき 0 を取る適合度である．

機械読解タスクの損失 L_{RC} は，正例に対する負の対数尤度により求める．

$$L_{RC} = -\frac{1}{N_{\mathrm{pos}}} \sum\nolimits_i r_i \left(\log p^1_{y_i^1} + \log p^2_{y_i^2}\right) \tag{9}$$

ここで，N_{pos} は正例の数，y_i^1 と y_i^2 は真の回答の開始・終了位置である．

3.8 テストプロセス

情報検索. 質問 q が与えられると，提案モデルはパッセージ集合 D に含まれる各パッセージ $x \in D$ に対して適合度 p^r を出力し，この値に基いて，k 個のランキングされたパッセージの集合 R_k を出力する．

機械読解. 質問 q が与えられると，提案モデルは検索結果集合 R_k に含まれる各パッセージ $x \in R_k$ について，動的計画法により $p^1_{t_1} p^2_{t_2}$ が最大となる回答範囲 (t_1, t_2) に対応する回答文字列を出力する．

大規模読解. 読解モジュールが出力した k 個の回答文字列について，検索モジュールの出力である $\exp(p^r/\tau)$ を重みとして重み付き多数決を行う．ここで，τ は多数決の重みを調整する温度パラメータである．

3.9 Telescoping

ニューラルネットワークを大規模テキスト集合に適用することは計算コストの面で難しい．そこで，複数個の情報検索モデルを連結し，後段のモデルはより少数の文書の再ランキングを行う telescoping 構成 [10] を導入する．提案モデルは汎用性を失うことなく telescoping を導入することが可能である．情報検索モジュールは，無関係なパッセージを除去することに焦点を絞ったキーワード検索モデルが出力した D のサブセットから，適合パッセージを正確に検索する．

4 実験

4.1 データセット

標準的な機械読解のデータセットである SQuAD [13] と，我々が作成した日本語ニュース記事に関するデータセット Jp-News により評価実験を行った（表 1）．Jp-News の特徴は，広範囲なトピックをカバーし，意

表 1: データセットの平均件数およびトークン数.

	SQuAD		Jp-News		
	train	dev	train	dev	test
記事数	442	48	4,000	500	500
質問数	87,599	10,570	66,073	8,247	8,272
パッセージ数	18,896	2,067	10,024	1,214	1,247
回答数	87,599	34,726	179,908	22,500	22,500
質問長	11.4	11.5	21.9	21.8	21.9
パッセージ長	140.3	144.5	181.4	176.2	177.7
回答長	3.5	3.3	4.3	4.5	4.2

図が明確な質問で構成されている点にある．本実験では test および development セットのパッセージ集合をそれぞれ D とした．また，各質問が参照する 1 パッセージのみを適合とし，その他は非適合と定義した．

4.2 評価指標

情報検索. 2 値の適合度判定用の指標である正解含有率（success at k; S@k）[4] と平均逆順位（mean reciprocal rank at k; M@k）[3] を用いた．

（大規模）機械読解. SQuAD と同様に完全一致（EM）とマクロ平均 F1 スコア（F1）[13] を用いた．日本語においては文字単位にて F1 を計算した．

4.3 ベースライン

情報検索. unigram, bigram に関する TF-IDF に基づく **BM25** を初期ランキング法として利用し，最新のニューラルモデルである **Duet** [12] と **Match-tensor** [7] を再ランキング法として利用した．

機械読解. 最新モデルである **R-NET** [17][1]，**Document Reader** [2]，**BiDAF** [14] を利用した．

大規模読解. Chen et al. [2] の手法に相当するアプローチとして，再ランキングなしの BM25 とシングルタスク学習を行った提案モデルの読解モジュールを単純結合したものをベースラインとした．また，同ベースラインに提案モデルの検索モジュールをシングルタスク学習で訓練したものを再ランキング法として追加したものについても比較を行った．

4.4 モデル詳細設定

前処理. トークナイザとして SQuAD 向けに Stanford CoreNLP [9]，Jp-News 向けに JTAG [5] を用いた．単語埋め込みについては 300 次元の fastText [1] を大文字・小文字を区別して英語・日本語の Wikipedia データから学習し，固定ベクトルとして利用した．

訓練. 10GPU で学習を行い，各 GPU は 60 サンプル（正例 30，負例 30）のミニバッチを処理した．負例として，対応する正例のパッセージに TF-IDF ベクトル空間で最も類似した 15 パッセージからランダムに選択して生成した．SGD（モメンタム値 0.9）により学習を行い，初期学習率を 1.0 とし，エポック毎に 0.9 ずつ減衰させた．ニューラルネットは Xavier uniform [6] にて初期化し，訓練時に各重みの指数移動平均（係数 0.99）を管理した．その他，$d = 100$，$c = 100$，ドロップアウト率 0.2，$\lambda = 1.0$ とした．

テスト. 各重みの指数移動平均を利用した．BM25 は初期ランキング法として 200 パッセージを検索し，提案モデルの検索モジュールは k（1 から 5）パッセージを検索した．その他，$\tau = 0.05$ とした．

[1]R-NET は 2018 年 12 月 18 日時点で SQuAD のリーダーボードの 1 位を獲得している．

表 2: 情報検索（再ランキング）タスク単体の評価結果.

	SQuAD		Jp-News			
	dev		dev		test	
re-ranker	S@1	M@5	S@1	M@5	S@1	M@5
(無し)	0.748	0.810	0.713	0.824	0.692	0.804
Duet	0.665	0.743	0.573	0.698	0.564	0.692
Match-tensor	0.732	0.791	0.725	0.821	0.704	0.806
提案（STL）	0.707	0.773	0.690	0.800	0.673	0.787
提案（MTL）	**0.811**	**0.863**	**0.753**	**0.842**	**0.737**	**0.830**

表 3: 機械読解タスク単体の評価結果.

	SQuAD		Jp-News			
	dev		dev		test	
RC model	EM	F1	EM	F1	EM	F1
BiDAF	67.7	77.3	76.9	88.1	77.3	88.3
Document Reader	69.5	78.8	75.9	87.6	76.2	87.8
R-NET	**71.1**	**79.5**	N/A			
提案（STL）	69.1	78.2	77.4	88.4	78.3	88.8
提案（MTL）	69.3	78.5	**78.0**	**88.8**	**78.8**	**89.2**

4.5 評価結果

マルチタスク学習（MTL）と telescoping 設定の有効性について評価を行った．ニューラルモデルについてはそれぞれ 5 回の試行を行った結果の平均を示す．

マルチタスク学習はシングルタスク学習よりも情報検索を精度面で改善するか？ 表 2 に，提案モデルの情報検索モジュールを再ランキング法として評価した結果を示す．提案モデルは，情報検索と機械読解で中間層を共有したマルチタスク学習を行うことで，全評価セットにおいて初期ランキング法である BM25 を有意に上回る精度を得た（t-test; $p < .001$）．その一方で，他の再ランキング法は訓練データ数の少なさが問題となり，BM25 を上回る精度を得られていない．興味深いことに，提案モデルの情報検索モジュールを通常のシングルタスク学習（STL）で訓練したものは BM25 の精度を大きく下回っている．この結果は，情報検索において，文書全体に適合度を与えて学習するよりも，文書のどの部分が質問に対する回答かを明確にして学習することが重要であることを示唆している．

マルチタスク学習はシングルタスク学習よりも機械読解を精度面で改善するか? 表 3 に標準的な機械読解タスク（回答を含む適合パッセージ 1 つが与えられる設定）における，提案モデルの機械読解モジュールの貢献を示す．マルチタスク学習を行った提案モデルはシングルタスク学習の場合に比べて，全評価セットにおいて EM および F1 指標で有意に上回った（Two-way repeated-measures ANOVA; $p < .05$）．本研究は機械読解単体の state-of-the-art 精度を実現することを目的にはしていないが，従来の大規模読解モデルで利用する Document Reader モデル [2] を含む最新手法と匹敵する精度を実現できることを示した．

telescoping 設定を利用した提案モデルはパイプライン法の大規模機械読解を精度面で改善するか? 表 4 に示す様に，全評価セットにおいて，提案モデルは state-of-the-art のアプローチ [2] であるパイプライン法に比べて有意に高い精度であった（t-test; $p < .001$）．具体的に，5 モデルのアンサンブルを取る場合は，SQuAD dev において EM で 7.9%，F1 で 8.2% 精度が向上していた．情報検索モジュールの改善がこの大きな精度向上をもたらしている．

なお，telescoping 設定の初期ランキング法としての BM25 について評価したところ，S@200 は SQuAD に

表 4: 大規模機械読解の評価結果．

単一モデル（5 試行の平均性能）							
		SQuAD		Jp-News			
		dev		dev		test	
提案 IR	提案 RC	EM	F1	EM	F1	EM	F1
(無し)	STL*	53.9	61.6	65.6	78.0	65.6	77.9
STL	STL	52.2	59.9	64.9	77.2	65.5	77.7
MTL	MTL	**60.0**	**68.1**	**69.5**	**81.7**	**70.6**	**82.7**
アンサンブルモデル（5 モデルの多数決）							
提案 IR	提案 RC	EM	F1	EM	F1	EM	F1
(無し)	STL*	56.6	63.6	68.2	79.7	67.9	79.4
STL	STL	56.3	63.2	68.8	80.2	68.8	80.4
MTL	MTL	**64.5**	**71.8**	**73.5**	**84.5**	**75.0**	**85.9**

* BM25+ re-ranker 無し ＋ 提案 RC は [2] に対応．

Question: Who contends that Luther did not intend to oppose the church?
Answer: Hans Hillerbrand
(a) Single-Task Learning
Hans Hillerbrand writes that Luther had no intention of confronting the church
(b) Multi-Task Learning
Hans Hillerbrand writes that Luther had no intention of confronting the church

図 2: 情報検索アテンションの可視化（回答付近のみ）

対して 0.991，Jp-News dev に対して 0.997 と，質問に無関係なパッセージを正確に除去できていた．

telescoping 設定を利用した提案システムは実用的な速度で動作するか？ SQuAD dev セットを用いてテストプロセスの実行時間について計測したところ，telescoping 設定を利用した場合（ニューラルネットは 200 パッセージを処理）は 1 質問あたり 1.5 秒，利用しない場合（2,047 パッセージを処理）は 1 質問あたり 17.1 秒であった．BM25 は 10 ミリ秒以下で高速に動作するため，telescoping 設定を利用した提案システム全体の計算量は，m は BM25 が出力する D のサブセットのサイズとした時 $\mathcal{O}(m)$ とみなせる．

その他の解析結果． 図 2 に，提案モデルの情報検索モジュールによる，入力質問に応じたパッセージの各単語に対するアテンションの値（β_t）を示す．マルチタスク学習を行った場合は回答フレーズの存在を強く注視するが，シングルタスク学習の場合は質問単語にのみ注視が働いた．また，SQuAD におけるクエリタイプ（when, where, why など）ごとに提案モデルの情報検索の精度を調査したところ，すべてのカテゴリで BM25 よりも有意に高い精度であった．さらに，情報検索モジュールが出力するパッセージ数 k の影響について調査したところ，SQuAD では $k = 1$，Jp-News では $k = 5$ が最も良い結果であった．SQuAD のように多くの記述が一度しか出現しないコーパスでは $k = 1$ が有効であり，Jp-News の様に同一の記述が複数回出現するコーパスでは大きな k の値を設定し重み付き多数決を取った方が精度が向上することがわかった．

5 おわりに

本研究は非構造テキスト集合を知識源とした質問応答を可能とする大規模機械読解に取り組んだ．

本研究の独自性． これまで情報検索と機械読解を単純に結合して取り組まれていた大規模機械読解 [2] において，中間層を共有することにより情報検索と機械読解のマルチタスク学習を行うニューラルネットワークモデルを新たに提案した．

本研究の重要性． 下記に貢献点を示す．大規模機械読解は新しい研究分野であり，これらの貢献は本分野の発展に大きく寄与すると考える．

- 情報検索と機械読解のマルチタスク学習の有効性を複数データセットにおいて示した．提案モデルは state-of-the-art のアプローチ [2] に対して，特に情報検索の精度を改善した．
- 提案したマルチタスク学習のアプローチは他の機械読解モデルにも検索層を追加することで適用可能であり，高い汎用性を持つ．
- 情報検索においてユーザの情報要求を満たす箇所を特定した学習が，文書単位の適合度を基にした学習よりも有効であることを初めて示した．
- telescoping [10] の導入によりモデルの汎用性を失うことなく実用的な速度での動作を可能にした．
- 日本語における機械読解の性能について初めて報告し，英語以外でも文法知識無しに機械読解ができることについて示した．

今後の課題． 本研究は回答範囲に基づく教師あり学習のみを扱っているため，TriviaQA [8] などの distant supervision [11] が必要なデータについてもマルチタスク学習のアプローチが有効であることを検証したい．

参考文献

[1] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching word vectors with subword information. *TACL*, 5:135–146, 2017.
[2] D. Chen, A. Fisch, J. Weston, and A. Bordes. Reading wikipedia to answer open-domain questions. In *ACL*, pp. 1870–1879, 2017.
[3] N. Craswell. Mean reciprocal rank. In *Encyclopedia of Database Systems*, p. 1703. 2009.
[4] N. Craswell. Success at n. In *Encyclopedia of Database Systems*, pp. 2875–2876. 2009.
[5] T. Fuchi and S. Takagi. Japanese morphological analyzer using word co-occurence -JTAG. In *COLING-ACL*, pp. 409–413, 1998.
[6] X. Glorot and Y. Bengio. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*, pp. 249–256, 2010.
[7] A. Jaech, H. Kamisetty, E. K. Ringger, and C. Clarke. Match-tensor: a deep relevance model for search. In *Neu-IR@SIGIR*, 2017.
[8] M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*, pp. 1601–1611, 2017.
[9] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky. The stanford corenlp natural language processing toolkit. In *ACL*, pp. 55–60, 2014.
[10] I. Matveeva, C. Burges, T. Burkard, A. Laucius, and L. Wong. High accuracy retrieval with multiple nested ranker. In *SIGIR*, pp. 437–444, 2006.
[11] M. Mintz, S. Bills, R. Snow, and D. Jurafsky. Distant supervision for relation extraction without labeled data. In *ACL/IJCNLP*, pp. 1003–1011, 2009.
[12] B. Mitra, F. Diaz, and N. Craswell. Learning to match using local and distributed representations of text for web search. In *WWW*, pp. 1291–1299, 2017.
[13] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. Squad: 100,000+ questions for machine comprehension of text. In *EMNLP*, pp. 2383–2392, 2016.
[14] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi. Bidirectional attention flow for machine comprehension. In *ICLR*, 2017.
[15] R. K. Srivastava, K. Greff, and J. Schmidhuber. Highway networks. eprint arXiv:1505.00387, 2015.
[16] O. Vinyals, M. Fortunato, and N. Jaitly. Pointer networks. In *NIPS*, pp. 2692–2700, 2015.
[17] W. Wang, N. Yang, F. Wei, B. Chang, and M. Zhou. Gated self-matching networks for reading comprehension and question answering. In *ACL*, pp. 189–198, 2017.